Deep Learning - Week 4

1. Using the Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$, what would be the bias-corrected first moment estimate after the first update if the initial gradient is 4?

   (a) 0.4

   (b) 4.0 Imp Change here

   (c) 3.6

   (d) 0.44

   Correct Answer: (a)
   Solution: In Adam, the first moment estimate is calculated as:

   $$m_t = \beta_1 * m_{t-1} + (1 - \beta_1) * g_t$$

   For the first update, $m_0 = 0$, so:

   $$m_1 = 0.9 * 0 + 0.1 * 4 = 0.4$$

   The bias-corrected first moment is:

   $$m_t^{corrected} = m_t/(1 - \beta_1^t)$$

   $$m_1^{corrected} = 0.4/(1 - 0.9^1) = 0.4/0.1 = 4$$

   Therefore, the bias-corrected first moment estimate after the first update is 4.

2. In a mini-batch gradient descent algorithm, if the total number of training samples is 50,000 and the batch size is 100, how many iterations are required to complete 10 epochs?

   (a) 5,000

   (b) 50,000

   (c) 500

   (d) 5

   Correct Answer: (a)
   Solution: Let's break this down step by step: 1) Number of batches per epoch = Total samples / Batch size = 50,000 / 100 = 500 batches 2) Number of iterations for 10 epochs = Number of batches per epoch * Number of epochs = 500 * 10 = 5,000 iterations

   Therefore, 5,000 iterations are required to complete 10 epochs.

3. In a stochastic gradient descent algorithm, the learning rate starts at 0.1 and decays exponentially with a decay rate of 0.1 per epoch. What will be the learning rate after 5 epochs?

(a) 0.09

(b) 0.059

(c) 0.05

(d) 0.061 Imp Change here

Correct Answer: (b)
Solution: The formula for exponential decay is:

$$\eta_t = \eta_0 * e^{-kt}$$

where η_0 is the initial learning rate, k is the decay rate, and t is the number of epochs.

Plugging in the values:

$$\eta_5 = 0.1 * e^{-0.1*5} \approx 0.1 * 0.60653 \approx 0.059$$

4. In the context of Adam optimizer, what is the purpose of bias correction?

(a) To prevent overfitting

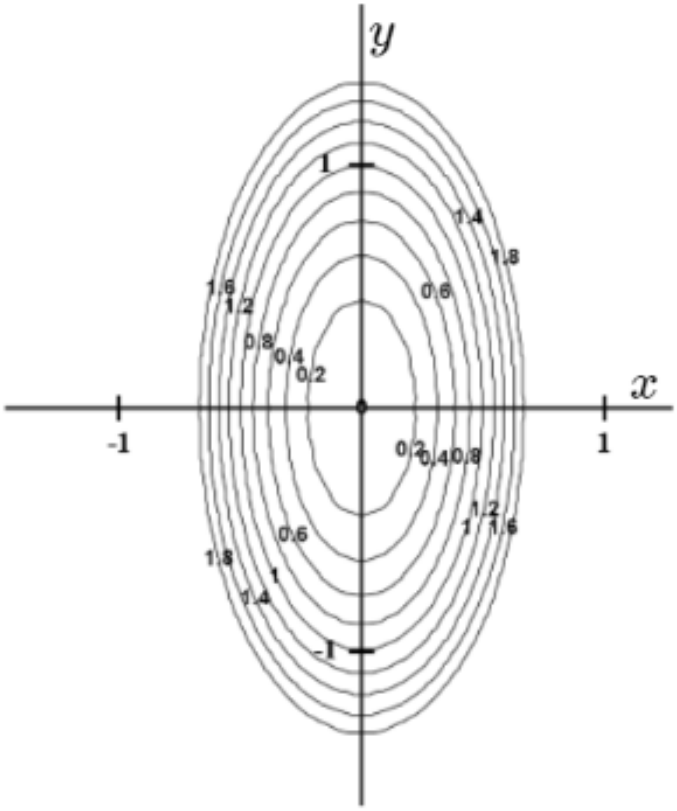
(b) To speed up convergence

(c) To correct for the bias in the estimates of first and second moments

(d) To adjust the learning rate

Correct Answer: (c)
Solution: In Adam optimizer, bias correction is used to correct for the bias in the estimates of first and second moments. This is particularly important in the early stages of training when the moving averages are biased towards zero due to their initialization.

5. The figure below shows the contours of a surface.



Suppose that a man walks, from -1 to +1, on both the horizontal (x) axis and the vertical (y) axis. The statement that the man would have seen the slope change rapidly along the x-axis than the y-axis is,

(a) True

(b) False

(c) Cannot say

Correct Answer: (a)

Solution: The given contour plot represents the function $f(x, y) = x^2 + 2y^2$. In a contour plot, the closeness of contour lines indicates the rate of change of the function. Since the contours are more closely spaced along the x-axis than along the y-axis, the function changes more rapidly in the x-direction. This means that a person walking from $x = -1$ to $x = 1$ would experience steeper slope changes compared to walking along the y-axis. Therefore, the statement that the slope changes more rapidly along the x-axis than the y-axis is **True**.

6. What is the primary benefit of using Adagrad compared to other optimization algorithms?

(a) It converges faster than other optimization algorithms.

(b) It is more memory-efficient than other optimization algorithms.

(c) It is less sensitive to the choice of hyperparameters(learning rate).

(d) It is less likely to get stuck in local optima than other optimization algorithms.

Correct Answer: (c)
Solution: The main advantage of using Adagrad over other optimization algorithms is that it is less sensitive to the choice of hyperparameters.

7. What are the benefits of using stochastic gradient descent compared to vanilla gradient descent?

(a) SGD converges more quickly than vanilla gradient descent.

(b) SGD is computationally efficient for large datasets.

(c) SGD theoretically guarantees that the descent direction is optimal.

(d) SGD experiences less oscillation compared to vanilla gradient descent.

Correct Answer: (a),(b)
Solution: SGD updates weight more frequently hence it converges fast. Since it is computationally faster than vanilla gradient descent, it works well for large datasets.

8. Select the true statements about the factor β used in the momentum based gradient descent algorithm.

(a) Setting $\beta = 0.1$ allows the algorithm to move faster than the vanilla gradient descent algorithm

(b) Setting $\beta = 0$ makes it equivalent to the vanilla gradient descent algorithm

(c) Setting $\beta = 1$ makes it equivalent to the vanilla gradient descent algorithm

(d) Oscillation around the minimum will be less if we set $\beta = 0.1$ than setting $\beta = 0.99$

Imp Change here

Correct Answer: (a),(b)(d)
Solution: Let's analyze the statements about the factor β used in the momentum-based gradient descent algorithm:

Momentum-based Gradient Descent: The momentum-based gradient descent algorithm updates the weights using the following rule:

$$v_{t+1} = \beta v_t + (1 - \beta)\nabla w_t$$

$$w_{t+1} = w_t - \eta v_{t+1}$$

where:
$-v_t$ is the velocity (momentum term).
$-\beta$ is the momentum factor.
$-\nabla w_t$ is the gradient of the loss with respect to the weight at time t.
$-\eta$ is the learning rate.

Setting $\beta = 0.1$ allows the algorithm to move faster than the vanilla (plain) gradient descent algorithm: - When β is set to a small positive value like 0.1, the algorithm incorporates some momentum, which can help accelerate convergence by navigating more effectively through shallow regions of the loss surface. This statement is generally true.

Setting $\beta = 1$ makes it equivalent to vanilla gradient descent algorithm: - If $\beta = 1$, the velocity term v_{t+1} would solely depend on the previous velocity v_t and would not incorporate the current gradient ∇w_t. This effectively stalls the learning process. However, the claim that it makes it equivalent to vanilla gradient descent (which does not use momentum) is incorrect. Vanilla gradient descent updates weights purely based on the gradient without momentum.

Setting $\beta = 0$ makes it equivalent to vanilla gradient descent algorithm: - When $\beta = 0$, the velocity term v_{t+1} is directly proportional to the current gradient ∇w_t. This reduces the momentum-based gradient descent to the plain gradient descent update rule. Thus, this statement is true.

Oscillation around the minimum will be less if we set $\beta = 0.1$ than setting $\beta = 0.99$: - Higher values of β (close to 1) result in more momentum, which can cause larger oscillations around the minimum due to the higher inertia. A lower value of β like 0.1 results in less momentum, leading to reduced oscillations. Therefore, this statement is true.

9. What is the advantage of using mini-batch gradient descent over batch gradient descent?

   (a) Mini-batch gradient descent is more computationally efficient than batch gradient descent.

   (b) Mini-batch gradient descent leads to a more accurate estimate of the gradient than batch gradient descent.

   (c) Mini batch gradient descent gives us a better solution.

   (d) Mini-batch gradient descent can converge faster than batch gradient descent.

   Correct Answer: (a),(d)
   Solution: The advantage of using mini-batch gradient descent over batch gradient descent is that it is more computationally efficient, allows for parallel processing of the training examples, and can converge faster than batch gradient descent.

10. In the Nesterov Accelerated Gradient (NAG) algorithm, the gradient is computed at:

    (a) The current position

    (b) A "look-ahead" position

    (c) The previous position

    (d) The average of current and previous positions

    Correct Answer: (b)
    Solution: In NAG, the gradient is computed at a "look-ahead" position. This look-ahead position is determined by applying the momentum step to the current position.

This allows the algorithm to have a sort of "prescience" about where the parameters are going, which can lead to improved convergence rates compared to standard momentum.